IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

Smart Hub

Samruddhi Shyam Jade[1], Rohan Dhananjay Gargade[2], Asavari Bandu Nawse[3], Omkar Sunil jadhav[4], Prof. Mrs. P. N. Kale[5]
Students, Department of Computer Technology[1,2,3,4]
Lecturer, Department of Computer Technology[5]
Sou. Venutai Chavan Polytechnic, Pune, Maharashtra, India

Abstract: *SmartHub is an advanced all-in-one productivity platform designed to streamline daily digital activities by integrating multiple essential tools into a single, unified dashboard. The system addresses the common challenge of managing and switching between various applications for tasks such as note-taking, reminders, and learning by providing a centralized and efficient workspace. Built on a modern decoupled architecture, SmartHub utilizes a React (Vite) frontend for a fast and responsive user interface, a Node.js and Express backend for handling APIs, and PostgreSQL with Prisma ORM for robust and scalable data management. The platform incorporates intelligent features such as a water reminder system powered by scheduled background jobs using node-cron, and an AI Quiz Hub that dynamically generates questions using AI APIs, enhancing user engagement and personalized learning.*

Keywords: Artificial Intelligence, Generative AI, Natural Language Processing, Intelligent Tutoring System

I. INTRODUCTION

SmartHub is a modern, intelligent productivity platform designed to simplify and enhance the way users manage their daily digital tasks. In today's fast-paced environment, individuals often rely on multiple applications for activities such as note-taking, reminders, learning, and personal tracking, which leads to inefficiency and fragmentation. SmartHub addresses this problem by bringing all essential tools into a single, unified dashboard, creating a seamless and organized user experience. The platform integrates features like an intelligent water reminder system, AI-powered quiz generation, and a markdown-based notes workspace, making it both practical and innovative. Developed using a scalable and decoupled architecture with React (Vite), Node.js, Express, and PostgreSQL, SmartHub ensures high performance, flexibility, and reliability. By combining automation, artificial intelligence, and user-centric design, SmartHub aims to improve productivity, save time, and provide a smarter way to manage everyday tasks.

II. LITERATURE REVIEW

The development of SmartHub is supported by extensive research in the fields of web-based productivity systems, modern full-stack development, and AI-driven applications. Recent studies highlight the growing demand for intelligent productivity tools that can efficiently manage tasks, improve user engagement, and automate daily activities. For instance, a study on smart task management systems emphasizes that modern applications integrate features such as task prioritization, reminders, and real-time analytics to enhance productivity and user efficiency. These systems significantly improve workflow management and help users track and optimize their daily activities. research on cloud-native web applications demonstrates that technologies like React and Node.js are widely adopted for building scalable and high-performance systems. These technologies enable the development of responsive user interfaces and efficient backend processing through event-driven architectures.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

III. METHODOLOGY

A. SYSTEM OVERVIEW

SmartHub is developed using a structured and modular methodology based on a decoupled full-stack architecture, ensuring scalability, flexibility, and maintainability. The system is divided into four major components: the frontend, backend, database, and background services. The frontend is built using React (Vite), providing a fast and interactive user interface for seamless user experience. The backend is developed using Node.js and Express, which handles API requests, business logic, and communication between different components. PostgreSQL is used as the primary database for storing structured user data such as notes, reminder settings, and quiz information, while Prisma ORM simplifies database operations and ensures type safety.

Additionally, background job scheduling is implemented using node-cron to automate tasks such as sending reminders. This modular approach allows each component to function independently while communicating through well-defined APIs, making the system efficient and easy to scale.Additionally, background job scheduling is implemented using node-cron to automate tasks such as sending reminders. This modular approach allows each component to function independently while communicating through well-defined APIs, making the system efficient and easy to scale.

B. WORKING OF THE SYSYTEM

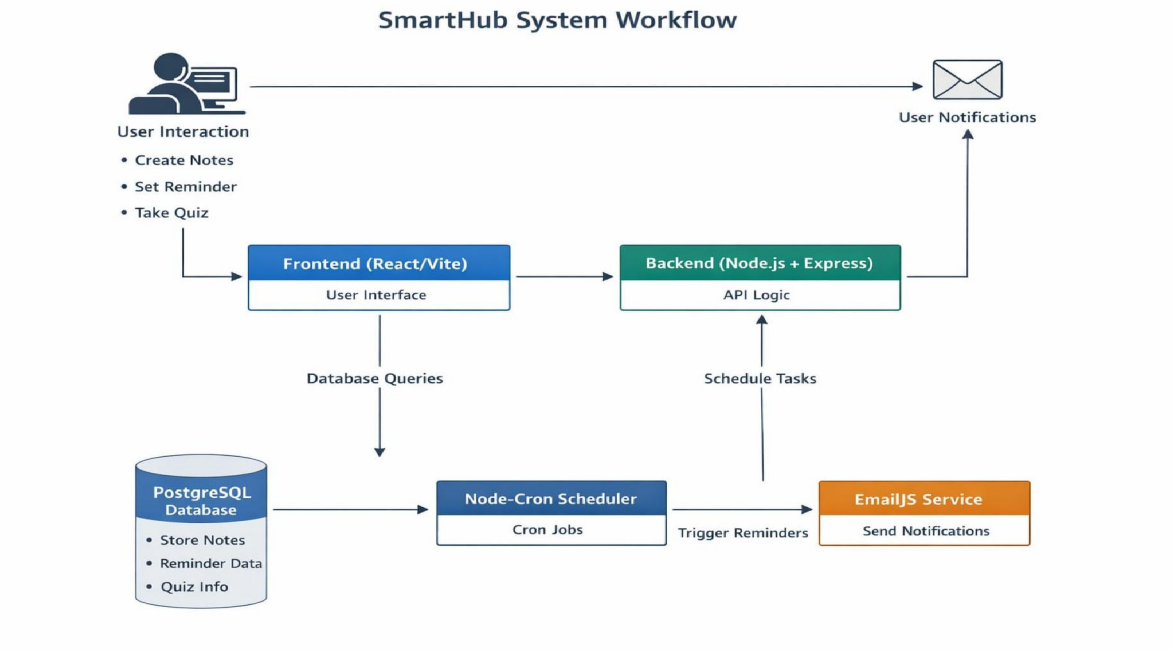


Fig. 1 Workflow of Smart Hub

The working of SmartHub begins when a user interacts with the frontend interface, such as creating notes, setting reminders, or accessing AI-based quizzes. These actions trigger API calls from the frontend to the backend using HTTP requests. The backend processes these requests, performs the required logic, and interacts with the PostgreSQL database through Prisma ORM to store or retrieve data. For features like the intelligent water reminder, user preferences are stored in the database, and node-cron schedules tasks that run at specified intervals to check and trigger reminders. When a reminder condition is met, the system sends notifications via EmailJS, ensuring the user receives updates even when not actively using the application. Similarly, the AI Quiz Hub communicates with external AI APIs to generate dynamic questions based on user input. All components work together in real-time, ensuring smooth data

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

flow, quick responses, and an efficient user experience, thereby making SmartHub a reliable and intelligent productivity system.

The working flow of SmartHub begins when a user logs into the system through the React (Vite)-based frontend interface, where all features are accessible via a centralized dashboard. Once authenticated using JWT, the user can perform actions such as creating notes, setting reminders, or accessing the AI Quiz Hub. Each user action triggers an HTTP request using Axios, which is sent to the Node.js and Express backend. The backend acts as the core processing unit, where it validates the request, applies business logic, and communicates with the PostgreSQL database through Prisma ORM to store or retrieve data.

For example, when a user sets a water reminder, the reminder preferences (time interval, start time, and frequency) are saved in the database. The node-cron scheduler running on the backend continuously monitors these stored schedules and executes background jobs at defined intervals. When the scheduled condition is met, the backend triggers the EmailJS service to send notification emails to the user, ensuring reminders are delivered even if the user is not actively using the application. Similarly, when a user selects a topic in the AI Quiz Hub, the backend sends a request to the AI API (such as Gemini), processes the generated questions, and sends them back to the frontend for display.

All components work in a synchronized manner where the frontend handles user interaction, the backend manages logic and processing, the database ensures persistent storage, and background services automate tasks. This continuous data flow ensures real-time updates, seamless user experience, and efficient execution of all functionalities, making SmartHub a highly responsive and intelligent productivity system.

The working of SmartHub follows a continuous and well-coordinated flow between user interaction, system processing, and automated services. Initially, the user accesses the platform through a secure login system, after which a personalized dashboard is loaded with all available modules such as Notes Workspace, Reminder System, and AI Quiz Hub. When the user performs any action—like creating a note—the frontend captures the input and sends it to the backend through API calls. The backend processes the request, validates the data, and stores it in the PostgreSQL database using Prisma ORM. Whenever required, the stored data is retrieved and displayed back to the user in real time, ensuring a smooth and responsive experience.

For AI-based features, such as the Quiz Hub, the system dynamically interacts with external AI services. When a user selects a topic or difficulty level, the frontend sends this data to the backend, which then forwards the request to the AI API. The generated questions are processed, formatted, and sent back to the frontend for display. This allows SmartHub to provide real-time, personalized learning content without storing large datasets internally.

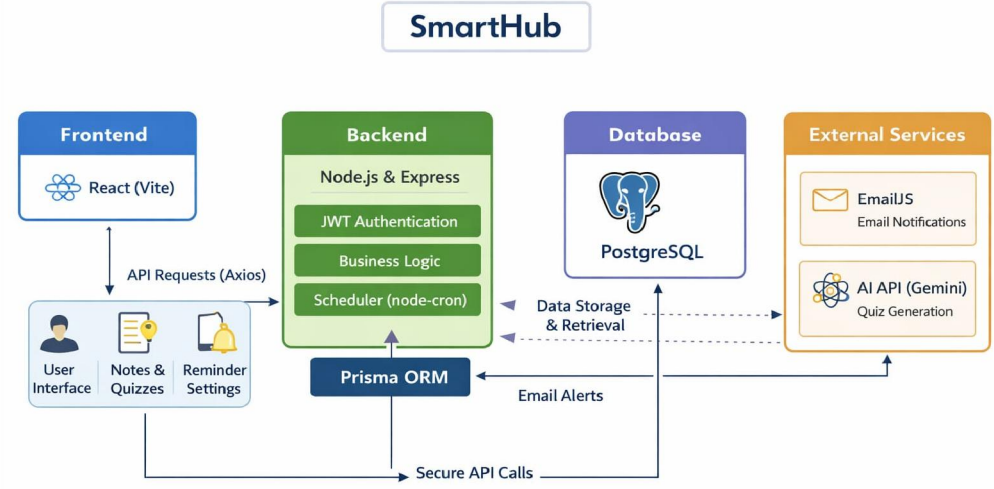


Fig. 2 System Architecture of Smart Hub

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

All these components work together to process user input, generate intelligent responses, and display the output efficiently, making the system scalable and user-friendly.

IV. ADVANTAGES OF THE PROPOSE SYSTEM

The SmartHub system offers numerous advantages by integrating multiple productivity tools into a single, unified platform, thereby reducing the need for users to switch between different applications. One of the key benefits is improved efficiency, as users can manage notes, reminders, and learning activities from a centralized dashboard. The system leverages a modern decoupled architecture, which enhances scalability, flexibility, and ease of maintenance, allowing independent development and deployment of frontend and backend components. The use of intelligent automation, such as node-cron for scheduling reminders, ensures that tasks are executed reliably even when the user is offline. Additionally, the integration of AI-powered features like dynamic quiz generation provides a personalized and engaging user experience. The use of PostgreSQL and Prisma ORM ensures secure, structured, and efficient data management, while JWT-based authentication enhances system security. Furthermore, the adoption of modern technologies like React (Vite) ensures fast performance and a smooth user interface. Overall, SmartHub improves productivity, saves time, enhances user engagement, and provides a scalable and future-ready solution for managing daily digital activities.

SmartHub also supports easy customization, allowing users to set preferences such as reminder frequency, topics for quizzes, and note organization according to their needs. The system is designed with modularity, which makes it easier to add new features like mobile app integration, push notifications, or additional AI tools in the future. Furthermore, the use of efficient technologies ensures low response time and high performance, even with multiple concurrent users.

V. LIMITATIONS

Despite its many advantages, the SmartHub system has certain limitations that may affect its performance and usability. One of the primary limitations is its dependency on internet connectivity, as most features such as AI quiz generation, email notifications, and data synchronization require an active connection to function effectively. The system also relies on third-party services like EmailJS and AI APIs, which may introduce issues such as API rate limits, service downtime, or additional costs. Another limitation is related to scalability of background jobs, where handling a large number of scheduled reminders simultaneously can increase server load and affect performance if not properly optimized.

Additionally, since SmartHub is primarily a web-based application, it may lack the native experience and performance optimization of mobile applications, which could impact user engagement on mobile devices. The system may also face data security and privacy challenges if not continuously updated with the latest security practices. Furthermore, AI-generated content may sometimes lack accuracy or consistency, which can affect the reliability of the quiz feature. Finally, as the system grows, managing large volumes of data in PostgreSQL may require further optimization techniques such as indexing and caching to maintain performance. Overall, while SmartHub is efficient and scalable, addressing these limitations is essential for improving its robustness and user experience.

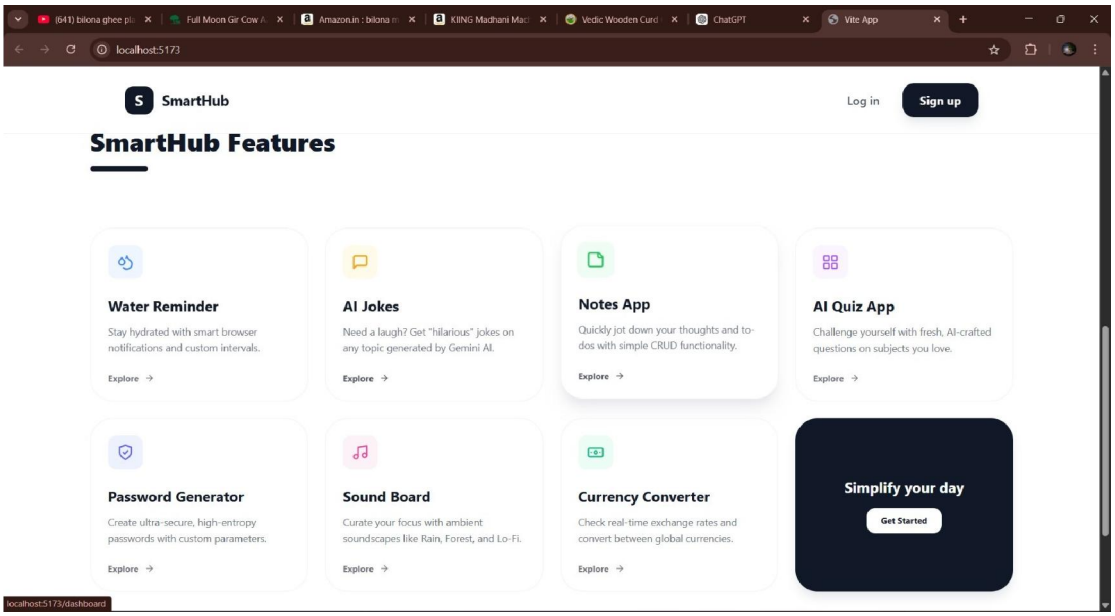
V. RESULTS AND DISCUSSION

The implementation of SmartHub demonstrates a successful integration of multiple productivity features into a single, unified platform, resulting in improved user efficiency and convenience. The system effectively performs core functionalities such as note management, intelligent reminder scheduling, and AI-based quiz generation with smooth and responsive interactions. The use of a decoupled architecture has shown positive results in terms of scalability, maintainability, and performance, as each component operates independently while maintaining seamless communication through APIs. The background scheduling using node-cron ensures that reminders are triggered reliably even when the user is inactive, validating the system's ability to handle automated tasks efficiently.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

Furthermore, the AI Quiz Hub has proven to enhance user engagement by generating dynamic and personalized content, making the platform more interactive and beneficial for learning purposes. The integration of PostgreSQL with Prisma ORM provides structured and efficient data handling, while JWT-based authentication ensures secure access to the system. However, during testing, certain challenges such as dependency on external APIs and performance considerations under heavy load were observed, indicating areas for future improvement. Overall, the results indicate that SmartHub is a robust, efficient, and user-friendly system that successfully meets its objective of simplifying daily digital tasks while offering intelligent and automated features.

1. HOME INTERFACE

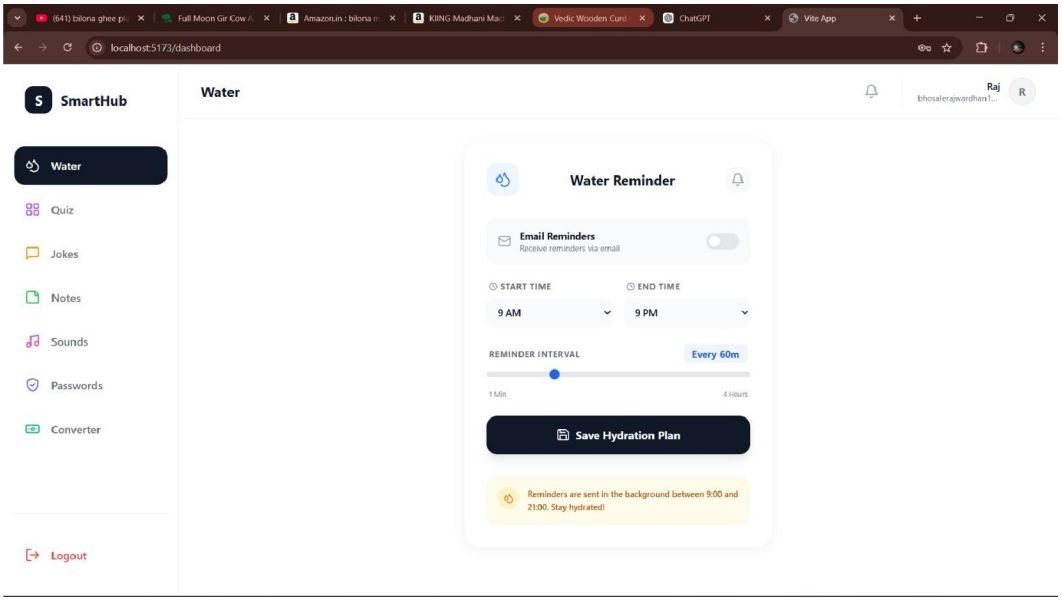


The SmartHub Features dashboard provides a centralized view of all available tools within the platform. It includes modules such as Water Reminder, AI Jokes, Notes App, AI Quiz, and more. Each feature is presented in a card-based layout for easy navigation and accessibility. This interface enhances user experience by offering a clean, organized, and intuitive design.

2. The Water Reminder module

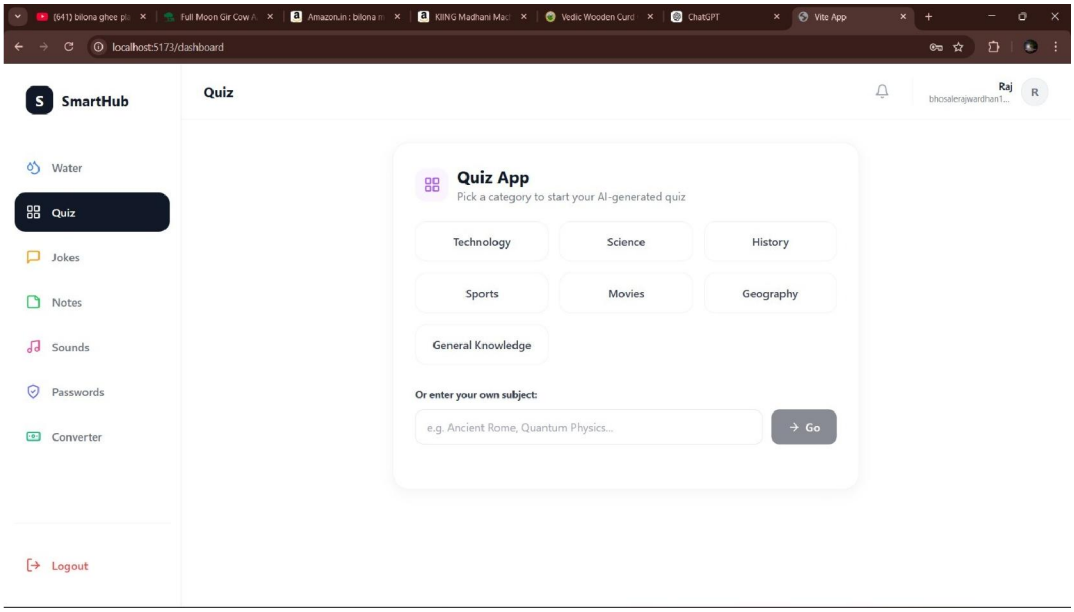
The Water Reminder module allows users to set personalized hydration schedules based on their daily routine. Users can define start time, end time, and reminder intervals with ease. The system ensures reminders are sent automatically in the background using scheduled jobs. This feature promotes healthy habits by keeping users consistently hydrated.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2



3. AI Quiz Module:

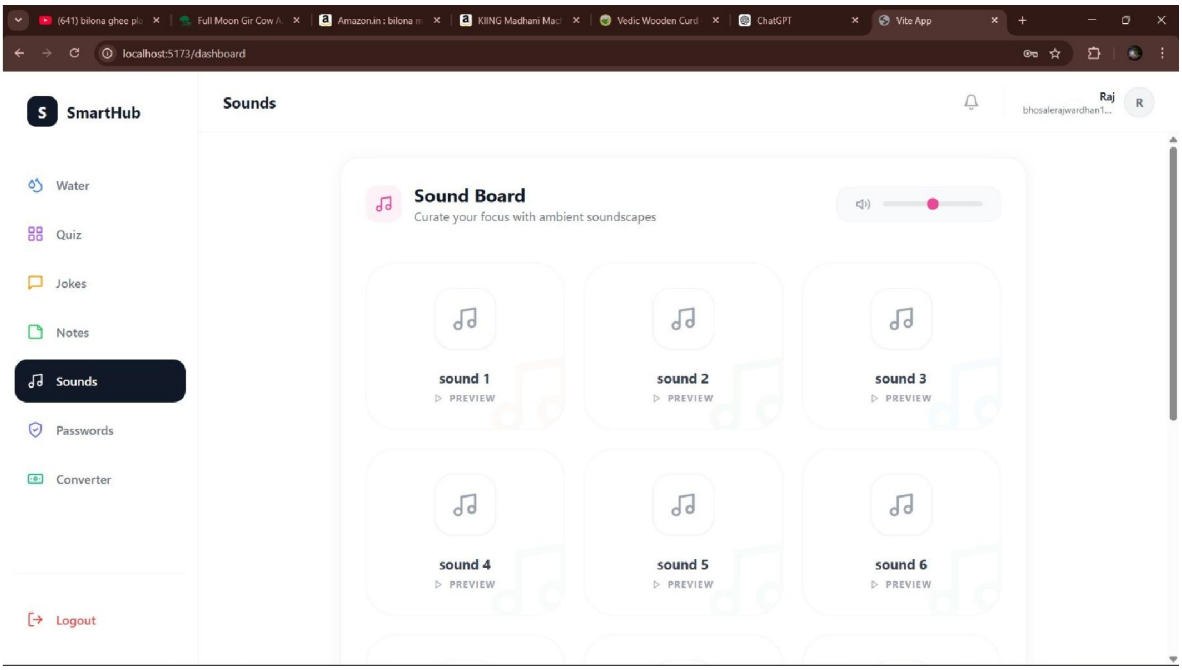
The AI Quiz module enables users to generate quizzes dynamically based on selected categories or custom topics. It leverages AI to create engaging and relevant questions in real time. The interface provides multiple subject options for quick access and interaction. This module enhances learning by offering an intelligent and personalized quiz experience.



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

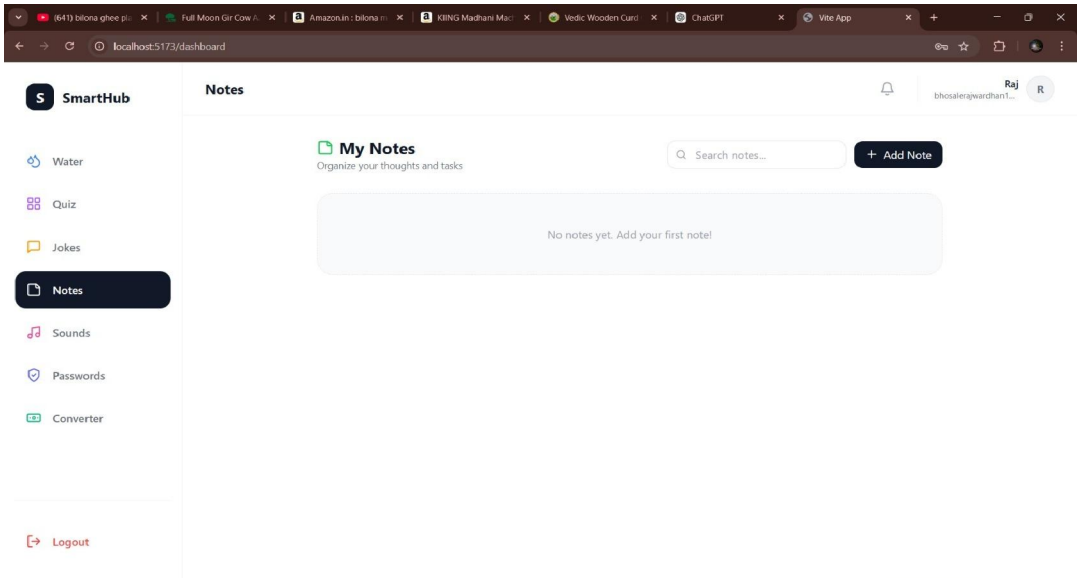
4. Sound Board Module:

The Sound Board module provides ambient sounds such as rain, forest, and lo-fi music to improve user focus. Users can preview and control sound playback with adjustable volume settings. The clean layout ensures easy selection of different sound options. This feature helps in creating a productive and distraction-free working environment.



5. Notes Workspace Module:

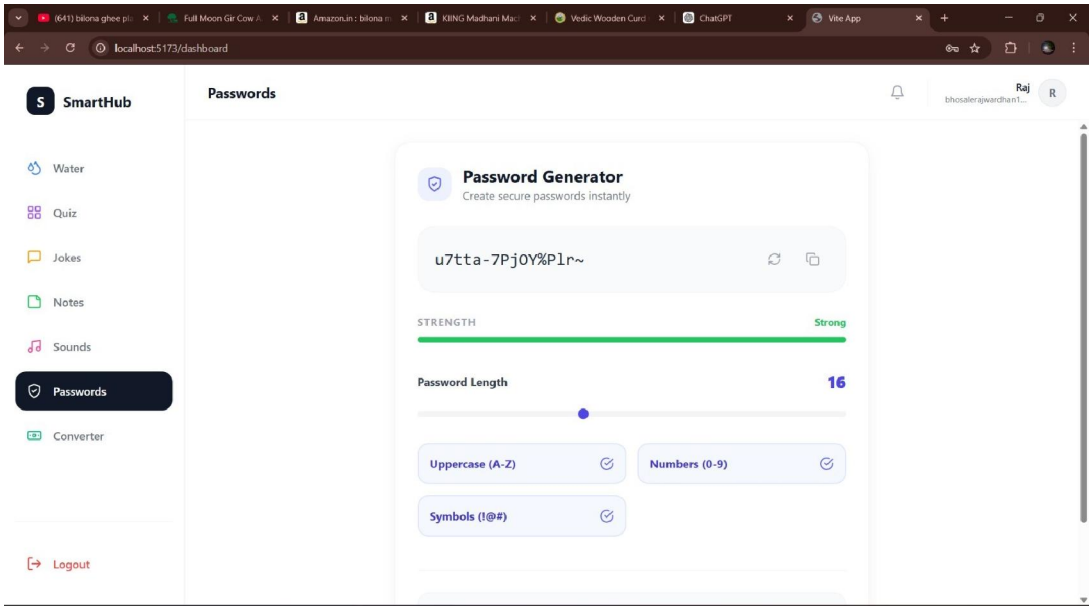
The Notes Workspace allows users to create, manage, and organize their personal notes efficiently. It supports basic CRUD operations with a simple and user-friendly interface. Users can search and add notes quickly, ensuring better information management. This module improves productivity by keeping important thoughts and tasks well organized.



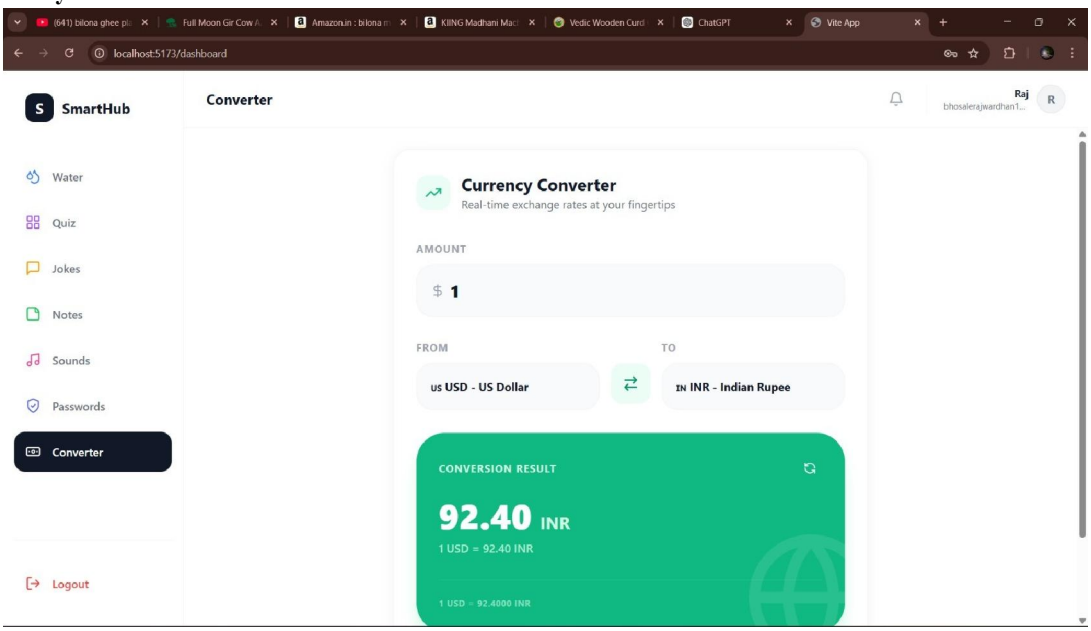
IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

6. Password Generator Module:

The Password Generator module helps users create strong and secure passwords instantly. It allows customization based on length, uppercase letters, numbers, and symbols. The system also provides a strength indicator to ensure password reliability. This feature enhances security by encouraging the use of complex passwords.



7. Currency Converter Module:



The Currency Converter module provides real-time conversion between different global currencies. Users can input an amount and select currencies to get instant results. The interface is simple and displays accurate exchange rates clearly. This feature is useful for users dealing with international transactions or financial planning.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 6, March 2026
Impact Factor: 8.2

VI. CONCLUSION

SmartHub successfully demonstrates the development of an intelligent and integrated productivity platform that combines multiple daily-use tools into a single, efficient system. The project effectively addresses the problem of managing multiple applications by providing a unified dashboard for features such as note management, reminder scheduling, AI-based quizzes, and utility tools. By using a modern decoupled architecture with technologies like React (Vite), Node.js, Express, PostgreSQL, and Prisma ORM, the system ensures high performance, scalability, and maintainability. The integration of automation through node-cron and AI capabilities further enhances the functionality and user experience of the platform.

Moreover, SmartHub not only improves productivity but also promotes better organization, learning, and time management. The system has been designed with a focus on usability, security, and future scalability, making it suitable for real-world applications. Although certain limitations such as dependency on external services and scalability challenges exist, they can be addressed in future enhancements. Overall, the project achieves its objectives and provides a strong foundation for developing advanced, intelligent, and user-centric productivity solutions.

VII.ACKNOWLEGMENT

We would like to express our sincere gratitude to all those who supported and guided us throughout the development of the SmartHub project. We are especially thankful to our mentors and faculty members for their valuable guidance, encouragement, and technical insights, which played a crucial role in the successful completion of this project. We also extend our appreciation to our institution for providing the necessary resources and a conducive environment for learning and development. Additionally, we would like to thank our team members for their continuous collaboration, dedication, and hard work. Finally, we are grateful to our family and friends for their constant support and motivation, which helped us overcome challenges and successfully complete this project.

REFERENCES

[1] Banks, A., & Porcello, E. (2020). Learning React: Modern Patterns for Developing React Applications. O'Reilly Media.
[2] Tilkov, S., & Vinoski, S. (2010). Node.js: Using JavaScript to Build Scalable Network Applications. IEEE Internet Computing.
[3] Kleppmann, M. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media.
[4] Fowler, M. (2018). Refactoring: Improving the Design of Existing Code. Addison-Wesley Professional.
[5] React (Vite) Documentation. Available at: https://react.dev/
[6] Node.js Official Documentation. Available at: https://nodejs.org/en/docs/
[7] Express.js Framework Guide. Available at: https://expressjs.com/
[8] PostgreSQL Official Documentation. Available at: https://www.postgresql.org/docs/
[9] Prisma ORM Documentation. Available at: https://www.prisma.io/docs/
[10] Tailwind CSS Documentation. Available at: https://tailwindcss.com/docs/
[11] Axios HTTP Client Documentation. Available at: https://axios-http.com/